Persistence of Time Management Behavior of Students and Its Relationship with Performance in Software Projects

Häkkinen, Joonas

ACM
2021-08-16

Häkkinen , J , Ihantola , P , Luukkainen , M , Leinonen , A & Leinonen , J 2021 , Persistence of Time Management Behavior of Students and Its Relationship with Performance in Software Projects . in ICER 2021: Proceedings of the 17th ACM Conference on International Computing Education Research . ACM , New York, United States , pp. 92-100 , ACM Conference on International Computing Education Research , 16/08/2021 . https://doi.org/10.1145/3446871.3469767

http://hdl.handle.net/10138/333457
https://doi.org/10.1145/3446871.3469767

acceptedVersion

Persistence of Time Management Behavior of Students and Its Relationship with Performance in Software Projects

Joonas Häkkinen
joonas.hakkinen@helsinki.fi
University of Helsinki
Helsinki, Finland

Petri Ihantola
petri.ihantola@helsinki.fi
University of Helsinki
Helsinki, Finland

Matti Luukkainen
mluukkai@cs.helsinki.fi
University of Helsinki
Helsinki, Finland

Antti Leinonen
antti.leinonen@helsinki.fi
University of Helsinki
Helsinki, Finland

Juho Leinonen
juho.2.leinonen@aalto.fi
Aalto University
Espoo, Finland

ABSTRACT

Teachers often preach for their students to start working on assignments early. There is even a fair amount of scientific evidence that starting early is beneficial for learning. In this work, we investigate students' time management behavior in a second-year project-based software engineering course. In the course, students work on a software project in small groups of four to six students. We study time management from multiple angles. Firstly, we conduct an exploratory factor analysis and study how different time management related behavioral metrics are related to one another, for example, whether individual students' time management practices in the second-year group project-based course are similar to their earlier time management practices in first-year courses where students work on assignments individually. Understanding how students' previous time management behavior is manifested in later project-based courses would be beneficial when designing project-based education. Secondly, we study whether students' time management practices affect the peer-review scores they get from their group members. Lastly, we explore how time management affects course performance. Our findings suggest that time management behavior, even from courses taken in the past, can be used to predict how students perform in future courses.

CCS CONCEPTS

• **Social and professional topics** → **Computing education**.

KEYWORDS

time management, group work, project-based learning, deadline driven, earliness

ACM Reference Format:
Joonas Häkkinen, Petri Ihantola, Matti Luukkainen, Antti Leinonen, and Juho Leinonen. 2021. Persistence of Time Management Behavior of Students and Its Relationship with Performance in Software Projects. In *Proceedings of the 17th ACM Conference on International Computing Education Research (ICER 2021), August 16–19, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3446871.3469767

ICER 2021, August 16–19, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8326-4/21/08…$15.00
https://doi.org/10.1145/3446871.3469767

1 INTRODUCTION

It is widely believed that starting assignments early is beneficial for academic performance. This belief is also backed to some extent by a body of academic studies (see eg. [6, 22, 25]). Despite that, a large number of students behave in a *deadline-driven* manner. To combat this behavior, one possible solution could be to have students work on their assignments in a group setting. Group work could influence students' time management positively as other students would depend on their actions, asserting peer pressure on them to do their tasks on time. Working in a group could also force students to schedule their time better in order to participate in group cooperation. On the other hand, group work could instigate students to not work as much on the assignments as they should since they could also depend on others, i.e. they might think that "someone else will do it so I don't have to".

In this paper, we study to what extent time management, that is, the timing-related working behavior in a programming course where students are taking individual assignments translates to group work in an agile software engineering course. We are interested in whether deadline-driven behavior in one context translates to another kind of setting, one that values highly frequent cooperation, and where group members work toward a common goal and are accountable to one another.

In any group work, cooperation among group members is of vital importance. Agile software engineering [4] puts a specific emphasis on tight interaction within group members, and one practical enabler for this interaction is to use continuous integration [10], that is, to frequently integrate each developer's code to a common code base. The Large-Scale Scrum -model even uses the phrase "communicate in code" [17] to describe this working pattern.

Whereas in a programming course the preferred working habit in doing assignments might be "start early", in agile project work that is not enough. Instead, one should start early and additionally work continuously in sync with other team members throughout the project. Quite often the phrase "Commit early, commit often" [1] is used to describe this steady way of working with small incremental synchronized steps.

The other aspect that we study in this paper is to what extent this continuous working pattern is beneficial in a group work setting. We examine whether it is valued by peers and how it relates to the outcome of the whole group.

2 RELATED WORK

2.1 Time Management from the Perspective of Educational Psychology

In the context of learning, time management refers to how students behave with regards to time. Claessens et al. [7] define time management as "behaviours that aim at achieving an effective use of time while performing certain goal-directed activities". Beneficial time management habits can include, for example, spacing out work [8, 19] into study sessions instead of cramming or massed work, i.e. working in a single long study session, and starting work early before the deadline for that work [2, 11, 13, 21, 27].

Several survey instruments have emerged to assess time management. According to a literature survey by Claessens et al. [7] the Time Management Questionnaire (TMQ) [6], the Time Management Behavioral Scale (TMBS) [20] and the Time Structure Questionnaire (TSQ) [5] are perhaps the most common of them.

The Time Management Questionnaire (TMQ) consists of short-range planning (e.g., do you make a list of the things you have to do each day), time attitudes (e.g., do you feel you are in charge of your own time), and long-range planning (e.g., do you usually keep your desk clear of everything other than what you are currently working on) and time attitudes. The first two items are related to good study performance, whereas the long range planning correlates negatively with SAT score and GPA. Authors of the scale argue that this dimension may indicate an inability to tolerate complexity. Correspondingly, preference for disorganization in TMBS is related to good study performance.

In TMBS time management is divided between the following four constructs: Setting Goals and Priorities (e.g., breaks down tasks); Mechanics—Planning, Scheduling (e.g., carries a notebook); Perceived Control of Time (e.g., feels in control of time); and Preference for Disorganization (e.g., messy workspace). All the dimensions are typically related to good academic performance. Dimensions of TSQ are similar to TMQ and TMBS, namely they are sense of purpose, structured routine, present orientation, effective organization, and persistence.

2.2 Time Management from the Perspective of Digital Footprints

While early work in time management has mostly relied on different questionnaires [7], in computer science education, using log data to study time management behaviors of students has become increasingly popular. Much of the research related to time management in computer science courses has focused on studying the relationship between time management and performance. There is a lot of empirical support for the notion that starting early could be beneficial for learning as many studies have found that better performing students start work earlier compared to their more poorly performing peers [2, 11, 13, 21, 27].

One concern raised in prior work [11] is that the effect of time management on performance could be caused simply by "better" students both achieving higher scores in assessments and starting early. Thus, a lot of previous work has tried to control for performance when analyzing the relationship between time management and performance [11, 15, 19, 21], for example by studying the effects of time management separately for students with different scores. This has been done to establish whether improving students' time management skills could in turn improve their performance. Many studies have found time management to have an effect beyond well-performing students having better time management [9, 11, 21]. For example, Denny et al. [9] conducted an A/B study where they found that students are more likely to start small assignments earlier, so the intervention affected students' time management in the aggregate supporting the notion that interventions that affect time management can increase performance. Edwards et al. [11] only analyzed students who had variation in their performance, i.e. were not consistently getting either good or poor grades from the exercises. They found that when those students achieved good grades, they exhibited better time management behavior such as starting earlier. Similarly, Martin et al. [21] studied students who had at least one early (submitted at least a day before the deadline) and one late (submitted after the deadline) submission, and found that for these students, the assignments that were submitted early had higher scores and passed more instructor provided tests.

Because of the large number of studies pointing to starting early possibly being beneficial, many studies have examined interventions aimed at getting students to start work earlier [15, 16, 21]. Martin et al. [21] studied three different interventions: reflective writing assignments, schedule sheets, and email alerts. In the reflective writing assignments students wrote about how their time management behavior affected their performance in projects; in the schedule sheets, students had to plan their time management related to the project they were working on; the email alerts included information about students' progress on the project related to the ideal progress at that point in the course. They found that the email alerts had a statistically significant effect on students' time management and made students complete assignments earlier. Ilves et al. [15] studied different visualizations related to self-regulation. They found that the visualizations had an effect on how early students started working on assignments, and that especially visualizations that showed a comparison to peers worked well. Irwin and Edwards [16] studied an intervention inspired by mobile gaming where students had a limited amount of "submission energy" which was required to submit assignments. They found that in the course where submission energy was used, students started work earlier.

Perhaps the closest match to our research is the one conducted by Auvinen et al. [3] where they studied how time management in individually done assignments translates to the group work done within an introduction to web development course. Their key finding was that students who tend to start late may drag the whole team down more than what the active students can save. On the other hand, if not focusing on time behavior, teams with both low and high performing (better grades, not time behavior) students perform similar to those with only high performers. The observations were made in relatively small teams (three students in each team).

3 METHODOLOGY

3.1 Research Questions

Much of the previously listed log-based time management measures have no explicit connection to educational psychology. Moreover, the measures are typically taken for granted and it is unclear if they are actually manifesting the same underlying constructs. In addition, more work is needed to understand the landscape of measuring time-related behavior in student-driven software projects. Thus, our first research question is *whether a lower number of unobserved variables could explain the variance in often used variables such as first action, last action, median action, and consistency of work as observed from different contexts* (RQ1).

Based on analysis for RQ1 (see Section 4.1 for details), it turned out that the underlying constructs for log-based measures courses with individual exercises and from group work were separate. Understanding the relationship between these (latent) measures, that is, individual time management and time management in group works is important because of multiple reasons. First, such information can be used in team building [3]. Second, based on previous research, time management can be learned [12, 28] and understanding the connection between different aspects of it would help us to design better learning experiences. Thus follows our second research question: *what is the relationship between prior individual time management and time management in group works* (RQ2).

In the course, students evaluate both their own work and their peers' work. These evaluations were designed to help the teachers of the course in evaluating the students. However, it is interesting to study whether there is a relationship between these self and peer evaluations and time management during the project: for example, do students rate their peers more highly if they exhibit better time management? Thus, our third research question is: *what is the relationship between group work time management, and self and peer evaluations?* (RQ3).

Based on previous research we know that time management measured as a psychological construct [5, 6, 20] or via simple log data [11, 18, 19, 21] is related to study performance. We wanted to test whether this holds also for latent constructs identified in RQ1. Thus, our last research question is *how the latent time management measures identified in this research correlate with course performance* (RQ4).

3.2 Context and Participants

The study was conducted in a big research-oriented university in Northern Europe. Our starting point was the group of 128 students (mean age=28 years, std=7.5years, 26.6% female) who participated in a 7-week Software Engineering course in the Autumn of 2020. Out of these, 101 students also had data for the earlier programming courses (CS1a, CS1b, CS2). Thus, in our analysis, we focus on the 101 students who had data from both the past and current course contexts.

The course was targeted for the second academic year, but some students may take it in their third year as well. The course covered a range of topics from software requirements to design and quality assurance, putting a special emphasis on agile processes and methods. The last three weeks of the course contained project work. All the project groups had the same topic and were working for a teaching assistant who acted as a product owner.

Each project had 5 or 6 students in the role of a developer. Groups were working in one-week sprints [23], i.e. in one-week iterations, during each of which they were supposed to implement a set of features according to the wishes of the product owner. Each group member was supposed to work six hours per sprint. Students were encouraged to work over multiple days.

Groups were free to organize their work as they wished, but all groups were expected to follow a certain baseline, such as to have a product and a sprint backlog [23] for tracking progress, to have automated tests, and to use version control and continuous integration for source code management. All of these topics had been covered earlier in the first four weeks of the course. Groups could select the technologies freely but since the working time was quite limited, they were encouraged to select technologies that were familiar to all group members.

Group member selection was done in a random manner such that each group was ensured to have some hours of common free working time during each week. Students were asked to fill in this information when signing up for the project. Since the study was conducted during the COVID-19 pandemic, the groups worked online and were quite actively using online tools such as Telegram, Zoom, Google Meet and Discord for interaction.

In addition to the Software Engineering course, data was collected from earlier courses as well. Individual time management habits were collected from the courses Introduction to programming (CS1a), Advanced programming (CS1b) and Data structures and algorithms (CS2). Both of the CS1 courses last for 7 weeks and include a weekly set of roughly 20 to 30 programming assignments. Both of the courses follow a "hands-on" approach where the theory and exercises are woven together to the course material. Typically a student first reads a small chunk of the theory part and then immediately applies it to a handful of small assignments. Some weeks especially in the latter part of the CS1b course contain also bigger, more open-ended assignments. Students majoring in computer science start their university studies with the CS1 courses.

The other source of individual time management data is the CS2 course that students typically take in their second semester. The course lasts for 14 weeks and contains a weekly set of 5 to 6 algorithmic exercises. In contrast to exercises in the CS1 courses, these are a bit more difficult, and students typically spend more time to complete the assignments.

When programming on these courses, students use an IDE [26] that captures a timestamp each time they run their code or submit their answer for evaluation, providing very accurate information of the times when the students are working on the assignments. In collecting the data used in this research, we followed the ethical procedures of the university where the data was collected. All the courses with individual tasks used the same learning management system, where students' consent for using learning data for research was also asked.

3.3 Measures

Here, the introduction of the measures is divided so that the metrics discussed in Sections 3.3.1, 3.3.2, and 3.3.3 are from the software engineering project course and the metrics in Section 3.3.4 are from the pre-requisite CS1 and CS2 courses.

3.3.1 Project Time Management Data. For the project work, the time management related working behaviour of students was collected from the timestamps of the commits they made during the project work. The source control system used by all groups was Git. Student-wise averages over the three sprints were computed into the following variables: 1) **project_start** - time difference between the end of sprint and the first commit during the sprint, in days, 2) **project_mid** - as above but for the median commit, 3) **project_end** - as above but for the last commit, and 4) **project_days** - number of days when commits were made during the sprint.

3.3.2 Peer Reviews. The students reviewed themselves and each other by answering to the following questions on a scale from 1 (poor contribution) to 4 (good contribution) individually for each group member, including the respondent themselves: 1) how well was each group member present, 2) how much did each group member contribute to the project's results, 3) how well did each group member's behavior help make teamwork meaningful, and 4) how active was each group member in contributing to the project?

Using the answers, student-wise average grades were computed on a per-question basis and the resulting variables were named **peer_q1 ... peer_q4**, respectively. The answers from the self-evaluation were used as variables named **self_q1 ... self_q4**, respectively.

3.3.3 Course Performance. Each group project was also graded by course assistants. Most of the emphasis was put on the process, that is, things such as backlog management, test automation and continuous integration. The outcome, if the end result was according to the product owner's wish, got a smaller emphasis, contributing roughly only 20% to the group score. Projects were scored after each sprint, each contributing one third to the project final score. The project score is called **project grade** in our analysis.

The course exam is also included in the study. The exam had written-answer questions on the following topics: Scrum, requirements management, Lean principles, DevOps, code quality and refactoring. The exam was online and students were free to use all the course material during the exam. The exam questions were formulated to measure students' ability to apply and interpret the course theory in the context of small case studies. The exam points are called **exam grade** in our analysis.

3.3.4 Individual Time Management Data. In the CS1 and CS2 courses used for analyzing past time management behavior, students typically use an IDE when working on assignments. The IDE captured a timestamp when students ran their code or submitted their answer for evaluation. These timestamps were used to form a temporal behavior dataset for the CS1 and CS2 courses. The timestamps were grouped by week, since all assignments of a given week shared a single deadline. The timestamps were then sorted chronologically, and any timestamps captured after the deadline were discarded. Removal of timestamps recorded after the deadline was done due to some students using the first assignment of the course as a sandbox for the following weeks, and any code runs for the first exercise still captured a timestamp. The assignments could not be submitted for evaluation after the assignment deadline. Following the same fashion as in 3.3.1, the data was averaged over all course weeks on a per-student basis, namely: 1) **past_start** is the time difference between a week's assignment deadline and the first recorded timestamp from that week, in days, 2) **past_mid** is the same for the median recorded timestamp of the week, 3) **past_end** is the same for the last recorded timestamp of the week, and 4) **past_days** is the number of days when timestamps were recorded during the week. Here, the word week refers to a round of exercises having the same same deadline and may differ from a calendar week depending on the course.

3.3.5 Similar Metrics in Earlier Work. The time management related variables were selected to match with earlier research. Ilves et al. [15] and Leppänen et al. [19] studied the number of days students worked actively, which is similar to our project_days and past_days metrics. Leppänen et al. [19] found that poorly performing students worked on more days than better performing students. Regarding the metrics related to starting, middle point and ending work, for example Edwards et al. [11], and Irwin and Edwards [16] have used the time of first submission as a metric of time management behavior; while Edwards et al. [11], Martin et al. [21] and Irwin and Edwards [16] have used the time of last submission as a metric. The median submission/commit in our study is similar to these metrics, but instead of measuring the start or end time of students' work, it measures the middle point. Many of the prior studies have found that students who start and finish earlier perform better in the course [9, 11, 21].

3.4 Data Analyses

Our first research question was about identifying the dimensions of time-related behavior. The collected log data and self reported data consists of multiple variables. The objective in RQ1 is to identify whether a smaller number of unobserved (latent) variables would explain the variance in the observed variables. To do so, an exploratory factor analysis (of the variables related to commit data, peer reviews, and IDE data) with oblique (Oblimin) rotation was used to understand the different aspects of time management. The number of extracted factors (i.e., latent variables) was based on parallel analysis, and calculated with the JASP software (v. 0.14.1).

The factors identified in RQ1 turned out to be separate for the variables illustrating the past behavior and behavior in the project course. Thus, in order to find how previous time related behavior predicts time related behavior in the group work (RQ2), we looked at how the factors related to past performance predict the factors related to time behavior in group work. Multiple linear regression with interaction of variables was used for the prediction.

To study the relationship of group work time management with self and peer evaluations for RQ3, we built a regression model to predict self and peer evaluations based on factors related to group work time management.

Finally, to find out how the interplay between all the time management characteristics identified in RQ1 explain course performance (RQ4), we investigated how the exam grade and the project grade could be predicted with all the time related factors (measured in the past and during the project). Multiple linear regression with interactions were pruned down by using step-wise regression to optimize the Akaike information criterion (AIC) (as implemented in the R package called MASS::stepAIC).[1]

4 RESULTS

4.1 Measures of Time Management (RQ1)

Kaiser-Meyer-Olkin measure of sampling adequacy (KMO=.715), and Bartlett's test of sphericity ($\chi^2(120)$ = 2092.263, p<.001) verified the adequacy of the factor analysis. This yielded a five-factor solution, accounting for 73.5% of the variance. The solution is illustrated in Table 1. The corresponding sum-variables were calculated as unweighted means of the related variables and named as *peer review* (i.e., peer_q1-4), *self evaluation* (i.e., self_q1-4), *past lateness* (i.e., past_start, past_mid, past_end), and *project lateness* (i.e., project_start, project_mid, project_end). In this context, lateness means how close to the deadline the work was done, with higher values of lateness corresponding to working closer to the deadline. Interestingly, past_days (illustrating consistency of working in the past) was alone in Factor 5. Moreover, with the cut off point 0.4, the observed variable project_days (illustrating consistency of working in the project) did not load to any of the factors. Both were still decided to be used in the further analyses as individual variables called *past consistency* and *project consistency*.

Table 1: Factor Loadings

	F1	F2	F3	F4	F5	Uniq.
past_start		0.836				0.037
past_mid		0.974				0.005
past_end		0.984				0.067
past_days					0.970	0.005
project_days						0.779
project_start				0.558		0.582
project_mid				0.986		0.005
project_end				0.582		0.613
self_q1			0.684			0.531
self_q2			0.708			0.329
self_q3			0.510			0.788
self_q4			0.960			0.074
peer_q1	0.907					0.104
peer_q2	0.869					0.078
peer_q3	0.941					0.237
peer_q4	0.958					0.013

Because the distributions of self evaluation, peer review project grade, and the actual grade were heavily left skewed (most students rated themselves and others very highly), they were transformed by using Ordered Quantile (ORQ) transformation, as suggested by the bestNormalize R-package[2]. The following analyses were conducted with the transformed variables. Pearson correlations for all the variables, provided in Table 2 start our exploration into RQs 2-4.

[1]https://cran.r-project.org/web/packages/MASS/MASS.pdf
[2]https://cran.r-project.org/web/packages/bestNormalize/

4.2 Past Individual Time Management vs. Time Management in Teams (RQ2)

In RQ2 we investigated the connection between prior individual time management and time management in group work. Although from the correlations point of view, (Table 2), only the project lateness is related to consistency of individual work, we constructed the multiple linear regression models where past lateness, past consistency, and their interactions were used as independent variables. The only significant model was the one where project latenness was explained with past consistency (R^2=0.06, F(1,99)=6.407, p=0.01.)

4.3 Group Work Time Management vs. Self and Peer Evaluations (RQ3)

In RQ3 we investigated the connection between group work time management and self/peer evaluation. Based on the correlations (Table 2), these seems to be a connection between both the project commit log based measures and peer review. Self evaluation seems to be linked with consistency, but not with the lateness. Again, multiple linear regression models were constructed, and the results are summarized in Table 3.

4.4 Time Management and Performance (RQ4)

To find out how time management affects performance, three multiple linear regression models were constructed to predict project grade and exam grades separately. Firstly, the *simple* model includes all the independent variables listed in Table 2 (grades excluded as those are considered as dependent variables). Next, the *full* model includes all the variables used in the simple model and all of their pairwise interactions. Finally, the *log-data* model was the same as full model, but with variables related to self and peer evaluations removed. All the models models were pruned with step-wise regression, as explained in methods. The results are summarized in Table 4.

5 DISCUSSION

5.1 Insights Gained from the Exploratory Factor Analysis (RQ1)

We performed an exploratory factor analysis (EFA) to study whether the measures we collected related to past time management in individual courses, current time management in the project course, self-evaluation and peer reviews are related to some harder to observe latent constructs. Variables loading into the same factor in the factor analysis means that they might measure the same latent construct. In layman's terms, if a group of variables correlate with each other strongly, they are likely to load into the same factor. Prior work has studied log-based measures for analyzing students' time management in similar manner [11, 18, 21, 24]. For example, both Spacco et al. [24] and Leinonen et al. [18] studied students' time on task based on log data, and Edwards et al. [11] and Martin et al. [21] studied earliness of students' work based on log data.

Looking at how different variables loaded into different factors in the factor analysis (see Table 1), we see that the variables related to consistency of work (past_days and project_days), i.e. how many days students worked on average, were in their own factors and variables related to when work was done in individual

Table 2: Pearson correlations between all the factors used in RQ2 and RQ3, and performance measures

	1.	2.	3.	4.	5.	6.
1. past lateness						
2. past consistency	-0.36***					
3. project lateness	0.16	-0.25*				
4. project consistency	-0.07	0.04	-0.22*			
5. peer review	0.13	0.11	-0.20*	0.28**		
6. self evaluation	-0.07	0.04	-0.04	0.22*	0.34***	
7. project grade	-0.25*	0.20*	-0.16	-0.11	0.06	0.12

Table 3: Predicting self evaluation and peer review with current log data. Statistical significance of the regression coefficients are coded as following: *p<.1; **p<.05; ***p<.01. The related standard errors are marked inside the parenthesis.

	Dependent variable:	
	self eval.	peer review
	(1)	(2)
project lateness	0.024	−0.098
	(0.107)	(0.104)
project consistency	0.228**	0.254**
	(0.101)	(0.098)
Observations	101	101
R^2	0.052	0.115
Adjusted R^2	0.023	0.088
Residual Std. Error (df = 97)	0.989	0.955
F Statistic (df = 3; 97)	1.774	4.217***

courses (past_start, past_mid, past_end) and in the group work (project_start, project_mid, project_end) were in their own factors. Factors related to past time management and project time management did not load to the same factor and the correlation between the factors *project lateness* and *past lateness* was quite low (.16). The factor model separated lateness (or earliness) from effort. *Project and course lateness* were calculated as averages of when the work was started, when half of the work was done, and when it was finished. This means that those who work a lot throughout the project are classified in the middle (early start and late completion).

There could be a few possible explanations for why project lateness and previous lateness were seen as separate factors. Firstly, it is possible that students' time management changes over time as the individual course data was from the first year and the project data was from the second year of studies. It is possible, for example, that students learn to manage time better as their studies progress. Another possibility for this is that how students work time-wise is different in individual courses and group work, perhaps due to team members influencing each other's time management behavior [3]. Lastly, since the data sources were different (IDE logs for past behavior and Git commits for project work), it is possible that measurements related to time management made at different times by different means are unable to measure the underlying phenomenon, i.e. time management, well enough, or they may measure different aspects of it.

The time management variables are also interrelated within the contexts: for example, considering the project work, the first commit has to be before the median and the last commit; and these also partly affect the number of days worked (if e.g. all first, median, and last commit are on the same day, the number of days worked has to be one), which could partly explain why data from one context tended to load to the same factor.

Time management scales used in educational psychology have dimensions such as time attitudes in TMBS or perceived control of time in TMQ that do not have obvious counterparts in log data. The previous examples are actually close to self-efficacy. Moreover, survey based instruments can provide a more fine grained explanation of the time related behavior, as the distinction between setting goals vs. mechanical time management in TMBS. Thus, we recommend that computing education community should more often combine these survey based instruments with log data, and acknowledge that we ourselves failed to do so.

5.2 Relationship Between Past and Current Time Management (RQ2)

Analyzing the relationship between past time management and project time management for RQ2 (see Section 4.2), we found that of the related metrics, the only statistically significant relationship was between the factors *past consistency* and *project lateness*, and even then only a relatively weak one (R^2=0.06). Here, the relationship is such that if you consistently worked on more days in past individual courses, you start work earlier in the project (correlation between these was -.25, see Table 2). Interestingly, the factor *past lateness* did not have a statistically significant relationship with *project lateness*. This suggests that students' time management is, for some reason, different in the two courses.

5.3 Relationship of Current Time Management with Self and Peer Evaluations (RQ3)

In RQ3 (see Section 4.3), we pondered how current time management gathered from log data relates to peer reviews and self evaluations. Analysis showed that project consistency, i.e. on how many days students work, had a statistically significant relationship with both self evaluation and peer reviews. How close to the deadline the students work did not have a statistically significant impact on self evaluation or peer reviews. These results can be seen in Table 3,

Table 4: Comparison of models predicting course performance with time management behavior. Statistical significance of the regression coefficients are coded as following: *p<.1; **p<.05; ***p<.01. The related standard errors are marked inside the parenthesis.

	Dependent variable:					
	project grade			exam grade		
	(simple)	(full)	(log only)	(simple)	(full)	(log only)
past lateness	−.214**	−.257**	−.223**	−.201**	−.168	
	(.097)	(.106)	(.098)	(.093)	(.102)	
past consistency		.024			.088	.208**
		(.102)			(.111)	(.098)
project lateness	−.153	−.137	−.150		.072	
	(.099)	(.099)	(.100)		(.102)	
project consistency	−.196*	−.204**	−.161		−.016	
	(.101)	(.099)	(.099)		(.097)	
peer review		.147		.371***	.322***	
		(.105)		(.093)	(.101)	
self evaluation	.160	.171*				
	(.099)	(.100)				
peer review:project lateness		−.269***				
		(.088)				
past lateness:past consistency		−.152				
		(.110)				
peer review:past lateness					−.163	
					(.108)	
peer review:past consistency					−.250**	
					(.107)	
peer review:project consistency					−.181*	
					(.100)	
project lateness:past consistency					−.150	
					(.105)	
Observations	101	101	101	101	101	101
R^2	.117	.228	.093	.159	.256	.043
Adjusted R^2	.080	.160	.065	.142	.183	.034
F Statistic	3.187**	3.387***	3.320**	9.241***	3.487***	4.487**
	(df = 4; 96)	(df = 8; 92)	(df = 3; 97)	(df = 2; 98)	(df = 9; 91)	(df = 1; 99)

and they seem to be in line with the course guidelines, as students were encouraged to start working early and to divide the work over multiple days. Hence, both early and late starters probably work until the deadline day.

5.4 Relationship Between Time Management and Course Performance (RQ4)

It would seem that past time management behavior is related to course performance. Based on the multiple linear regression (see Table 4), *past lateness*, i.e. when students tended to start and finish course work in individual courses in the first year, can predict the grade students will get in the group projects. Similarly, *past consistency*, i.e. on how many days students worked in individual courses in the first year, can predict the grade they will get in the exam for the second-year software engineering course. Prior work on analyzing programming log data has also found evidence that log data can be used to predict students' success [18]. One possible explanation for why time management in first year courses can predict success in the second year course under study is that well performing students exhibit positive time management already in the first year courses and also perform well in the second year course.

The results from the multiple linear regression showed that the peer reviews were the best predictors of exam grade. We see that in the absence of peer reviews, log data collected from the project work (i.e. the factors *project lateness* and *project consistency*) did not predict exam grade. This seems to suggest, that at least if work is done in groups, log data collected from it cannot be used to detect struggling students who might need additional support to perform well in the exam. Although, it is also possible that the relatively short project length (three weeks) affected this, and more data over a longer period of time would be found more useful.

5.5 Limitations

There are some issues that may have an effect to the generalizability of the results, some of which have already been discussed. First, the IDE/past time management data was collected from two different types of courses. The CS1a and CS1b are quite similar in structure, consisting of weekly exercise sets of between 20 to 30 small assignments. Unfortunately this data was not available for all of the students (since they had taken a version of the course that had no deadlines for assignment submission) and we needed to use only the CS2 course for those students to form a past time management profile. The CS2 course differs structurally as it has only 5 to 6 weekly assignments, all of which are somewhat more difficult than the CS1 assignments. This means that the past time management profiles of students could be affected by which courses were used to build those profiles. On the other hand, averaging over different contexts may be the reason why measures related to past behavior were in some cases more useful than the measures from the current course.

Additionally, related to analyzing how past time management relates to future time management and performance, there is a possibility for a survivorship bias: only the students who attend the second-year course have data for the second-year course, and students who, for example, drop out of their studies are not included. Thus, it is possible that the results we present here hold only for a particular group of students, i.e. those who persist until the second-year software engineering course. Future work should study how the time management profiles of drop outs and students who persist in their studies differ.

We only looked at students as a single cohort and did not control for performance. Thus, it is possible that in our context, some of the results would be explained by better students also having better time management skills. However, prior work has found that time management has an effect on performance beyond simply better students both performing better and having better time management skills [11, 21], and there is no reason to believe that our context would be different in this regard.

For the project course, we looked at Git commit timestamps. All commits were deemed equal in the sense that we made no difference between qualities such as a commit's size (e.g. the number of lines of code), content or other such metrics. The course project included only three week-long sprints during each of which the students were required to work for six hours. The short overall duration and relatively low workload may have an effect on the results. Additionally, students were encouraged in the lectures to work over multiple days, which can have an effect on students' time management.

The used assessment criteria as a way to communicate feedback is known to have a huge effect on student behavior during a course [14]. For each sprint, students were given a checklist about the things that would affect their score. It is possible that these clear guidelines caused the fact that there is a rather small variation in group scores and due to that it was difficult to properly differentiate good groups from dysfunctional ones.

6 CONCLUSIONS

In this work, we studied students' time management behavior in a project-based software engineering course. We studied how time management in the project work is related to past behavior in individual courses, and how time management behavior affects course performance. Here, as conclusions we answer our research questions:

Can a lower number of unobserved variables explain the variance in often used variables such as first action, last action, median action, and consistency of work as observed from different contexts? The exploratory factor analysis found a lower number of underlying latent factors. The latent factors in our case were related to lateness of work (first, median, last submission/commit), consistency of work (number of days that had submissions/commits), and these were in different factors for the individual assignments and group project work. Additionally, variables related to self-evaluation and those related to peer reviews were in their own two factors.

What is the relationship between prior individual time management and time management in group works? Consistency in previous individual exercises predicted group project lateness. We assume the context from where time management variables are collected is important, however. In our case, course organization may have had a significant role in how students behave in light of these measures. As in the project, all the students were instructed to work consistently, the instructions may have masked more natural behavior (if such existed).

What is the relationship between group work time management, and self and peer evaluations? Self evaluations are less accurate than peer reviews, and consistency in project work can be used to predict the peer review. The explained variance is very low, however. More research is needed to understand if this due to individual behavior and group behavior being different, or if the behavior does not persist over time.

How do the latent time management measures identified in this research correlate with course performance? We found that time management behavior from past courses can be used to predict success in a future exam. Both when students completed work and on how many days students worked on could be used for this purpose. Additionally, we found that students' consistency in project work, i.e. on how many days they worked on, could be used to predict the grade they eventually get for the project. Interestingly, time management in past individual courses could be used to predict future group work based project grades.

Our study is novel in that time management was studied for students over a long period of time (first and second year courses). Additionally, our results support earlier work (e.g. [11, 18]) that has found a relationship between students' time management related behavior and their performance in the course. Prior work has established that time management is related to performance beyond simply "good" students also having better time management [11, 21], and that time management can be taught to students [12, 28]. Based on these together with our results that time management behavior in the first-year courses can be used to predict success in the second-year course, one actionable pedagogical insight from this work is adding explicit teaching of time management to introductory courses as this could lead to increased performance in both those and future courses.

REFERENCES

[1] Jeff Atwood. 2001. Check In Early, Check In Often. https://blog.codinghorror.com/check-in-early-check-in-often/

[2] Tapio Auvinen. 2015. Harmful Study Habits in Online Learning Environments with Automatic Assessment. In *2015 International Conference on Learning and Teaching in Computing and Engineering*. 50–57.

[3] Tapio Auvinen, Nickolas Falkner, Arto Hellas, Petri Ihantola, Ville Karavirta, and Otto Seppälä. 2020. Relation of Individual Time Management Practices and Time Management of Teams. 1–9. https://doi.org/10.1109/FIE44824.2020.9274203

[4] Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, and Dave Thomas. 2001. Manifesto for Agile Software Development. http://www.agilemanifesto.org/

[5] Michael J Bond and NT Feather. 1988. Some correlates of structure and purpose in the use of time. *Journal of personality and social psychology* 55, 2 (1988), 321.

[6] Bruce Britton and Abraham Tesser. 1991. Effects of Time-Management Practices on College Grades. *Journal of Educational Psychology* 83 (09 1991), 405–410. https://doi.org/10.1037/0022-0663.83.3.405

[7] Brigitte JC Claessens, Wendelien Van Eerde, Christel G Rutte, and Robert A Roe. 2007. A review of the time management literature. *Personnel review* (2007).

[8] Frank N Dempster. 1988. The spacing effect: A case study in the failure to apply the results of psychological research. *American Psychologist* 43, 8 (1988), 627.

[9] Paul Denny, Andrew Luxton-Reilly, Michelle Craig, and Andrew Petersen. 2018. Improving Complex Task Performance Using a Sequence of Simple Practice Tasks. In *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education* (Larnaca, Cyprus) *(ITiCSE 2018)*. Association for Computing Machinery, New York, NY, USA, 4–9. https://doi.org/10.1145/3197091.3197141

[10] Paul Duvall. 2007. *Continuous Integration: Improving Software Quality and Reducing Risk.* Addison-Wesley, Reading, Massachusetts.

[11] Stephen H Edwards, Jason Snyder, Manuel A Pérez-Quiñones, Anthony Allevato, Dongkwan Kim, and Betsy Tretola. 2009. Comparing effective and ineffective behaviors of student programmers. In *Proceedings of the fifth international workshop on Computing education research workshop*. 3–14.

[12] Wendelien Van Eerde. 2003. Procrastination at work and time management training. *The Journal of psychology* 137, 5 (2003), 421–434.

[13] James B. Fenwick, Cindy Norris, Frank E. Barry, Josh Rountree, Cole J. Spicer, and Scott D. Cheek. 2009. Another Look at the Behaviors of Novice Programmers. In *Proceedings of the 40th ACM Technical Symposium on Computer Science Education* (Chattanooga, TN, USA) *(SIGCSE '09)*. Association for Computing Machinery, New York, NY, USA, 296–300. https://doi.org/10.1145/1508865.1508973

[14] John Hattie and Helen Timperley. 2007. The power of feedback. *Review of educational research* 77, 1 (2007), 81–112.

[15] Kalle Ilves, Juho Leinonen, and Arto Hellas. 2018. Supporting self-regulated learning with visualizations in online learning environments. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. 257–262.

[16] Michael S. Irwin and Stephen H. Edwards. 2019. Can Mobile Gaming Psychology Be Used to Improve Time Management on Programming Assignments?. In *Proceedings of the ACM Conference on Global Computing Education* (Chengdu,Sichuan, China) *(CompEd '19)*. Association for Computing Machinery, New York, NY, USA, 208–214. https://doi.org/10.1145/3300115.3309517

[17] Craig Larman and Bass Vodde. 2016. *Large-Scale Scrum: More with LeSS.* Addison-Wesley, Reading, Massachusetts.

[18] Juho Leinonen, Leo Leppänen, Petri Ihantola, and Arto Hellas. 2017. Comparison of time metrics in programming. In *Proceedings of the 2017 acm conference on international computing education research*. 200–208.

[19] Leo Leppänen, Juho Leinonen, and Arto Hellas. 2016. Pauses and spacing in learning to program. In *Proceedings of the 16th Koli Calling International Conference on Computing Education Research*. 41–50.

[20] Therese H Macan, Comila Shahani, Robert L Dipboye, and Amanda P Phillips. 1990. College students' time management: Correlations with academic performance and stress. *Journal of educational psychology* 82, 4 (1990), 760.

[21] Joshua Martin, Stephen H Edwards, and Clfford A Shaffer. 2015. The effects of procrastination interventions on programming project success. In *Proceedings of the eleventh annual International Conference on International Computing Education Research*. 3–11.

[22] Nicolas Michinov, Sophie Brunot, Olivier Le Bohec, Jacques Juhel, and Marine Delaval. 2011. Procrastination, participation, and performance in online learning environments. *Computers & Education* 56, 1 (2011), 243 – 252. https://doi.org/10.1016/j.compedu.2010.07.025 Serious Games.

[23] Ken Schwaber. 2004. *Agile Project Management with Scrum.* Microsoft Press, Redmond, WA. http://my.safaribooksonline.com/9780735619937

[24] Jaime Spacco, Paul Denny, Brad Richards, David Babcock, David Hovemeyer, James Moscola, and Robert Duvall. 2015. Analyzing student work patterns using programming exercise data. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*. 18–23.

[25] Piers Steel. 2007. The nature of procrastination: a meta-analytic and theoretical review of quintessential self-regulatory failure. Psychol Bull 133: 65-94. *Psychological bulletin* 133 (02 2007), 65–94. https://doi.org/10.1037/0033-2909.133.1.65

[26] Arto Vihavainen, Thomas Vikberg, Matti Luukkainen, and Martin Pärtel. 2013. Scaffolding students' learning using test my code. In *Proceedings of the 18th ACM conference on Innovation and technology in computer science education*. 117–122.

[27] Salla Willman, Rolf Lindén, Erkki Kaila, Teemu Rajala, Mikko-Jussi Laakso, and Tapio Salakoski. 2015. On study habits on an introductory course on programming. *Computer Science Education* 25, 3 (2015), 276–291. https://doi.org/10.1080/08993408.2015.1073829

[28] Barry J Zimmerman. 2002. Becoming a self-regulated learner: An overview. *Theory into practice* 41, 2 (2002), 64–70.